(Parsing Decomposition) of comparing sequence strings)

DOCUMENT-IDENTIFIER: US 20010047376 A1

TITLE: Method for the manipulation, storage, modeling, visualization and quantification of datasets

--------- KWIC --------

Abstract Paragraph - ABTX (1):

There is described a method for manipulation, storage, modeling, visualization, and quantification of datasets, which correspond to target strings. A number of target strings are provided. An **iterative algorithm** is used to **generate comparison strings** corresponding to some set of points that can serve as the domain of an **iterative** function. Preferably these points are located in the complex plane, such as in and/or near the Mandelbrot Set or a Julia Set. These **comparison** strings are also datasets. The **comparison string is scored by evaluating a function having the comparison string and one of the plurality of target strings** as inputs. The score measures a relationship between a **comparison string and a target string**. The evaluation may be repeated for a number of the other target strings. The score or some other property corresponding to the **comparison string is used to determine the target string's** placement on a map. The target string may also be marked by a point on a visual display. The coordinates of the point corresponding to the **target string or properties of the comparison** string may be stored in memory, a database or a table. Mapped or marked points in a region of interest can be explored by examining a subregion with higher resolution. The points are analyzed and/or **compared** by examining, either visually or mathematically, their relative locations, their absolute locations within the region, and/or metrics other than location.

Current US Classification, US Secondary Class/Subclass - CCSR (1):

707/1

Summary of Invention Paragraph - BSTX (7):

[0005] The present invention is a method for manipulation, storage, modeling, visualization, and quantification of datasets, which correspond to target strings. A number of target strings are provided. An **iterative algorithm** is used to **generate comparison strings** corresponding to some set of points that can serve as the domain of an **iterative** function. Preferably these points are located in the complex plane, such as in and/or near the Mandelbrot Set or a Julia Set. These **comparison strings are also datasets or data sequences**. The **comparison string is scored by evaluating a function having the comparison string and one of the plurality of target strings** as inputs. The evaluation may be repeated for a number of the other target strings. The score measures a relationship between a **comparison string and a target string**. In measuring a similarity relationship, for example, a one-to-one **comparison may be performed between the numbers in the comparison string and the target string**. In this example, the **comparison string having the highest score is deemed most similar to the target string**. The score or some other property corresponding to the **comparison string is used to determine the target string's** placement on a map. The target string may also be marked by a point on a visual display. The coordinates of the point corresponding to the **target string or properties of the comparison** string may be stored in memory, a database or a table. Mapped or marked points in a region of interest can be explored by examining a subregion with higher resolution. The points are analyzed and/or **compared** by examining, either visually or mathematically, their relative locations, their absolute locations within the region, and/or metrics other than location.

Summary of Invention Paragraph - BSTX (8):

[0006] The method allows for many advantages over the prior art. The fingerprinting and visualization of an entire dataset, and missing values are easily accommodated. The computational requirements are lower for this method because the mapping time increases only linearly with the size of the dataset and the number of datasets. The current data does not need to be remapped when new datasets are added to the map. The number of datasets that can be mapped and **compared** is unlimited. The map space of a region, such as the Mandelbrot or a Julia Set, is predetermined, fixed, and highly studied. The fact that the map is fixed and predetermined, along with the fact that data can be added to the map without recalculating the points already mapped, means the present invention can store this data in memory, a database or a table. Models of the datasets, or the **comparison strings, are created** in the mapping process.

These
features allow this method to be used not only in visualization and quantification of large datasets, but also for intelligent storage and modeling of such datasets.

Brief Description of Drawings Paragraph - DRTX (4):
[0009] FIG. 2 is a flow chart of the operational steps for an **iterative algorithm** and processing which provides a comparison string.

Detail Description Paragraph - DETX (4):
[0012] Starting with FIG. 1A, the method starts (step 101) by providing a set of M such target strings of length N* (step 103). A region R is selected (step 104) that can serve as the domain of an iterative function. The **iterative algorithm** calculates the comparison string from a point p in some region R. Preferably, the region R is in the complex plane corresponding to the area in and around the Mandelbrot Set. Although the Mandelbrot Set is used in the preferred embodiment of the present invention, other sets, such as Julia Sets, may also be used. Using this iterative method, every point within the Mandelbrot Set can be made to correspond to a data sequence of arbitrary length. Because the Mandelbrot Set is made up of an infinite number of points, the method allows any number of datasets containing any number of values to be
compared by mapping the datasets to points in or near the Mandelbrot Set.

Detail Description Paragraph - DETX (6):
[0014] A **comparison** string of length N is also provided (step 107) by using an **iterative algorithm**. The **comparison string is also a data string and may be**
of any length relative to the target string. FIG. 2 shows the steps involved in the **iterative algorithm to generate the comparison string** of length N provided in step 107 of FIG. 1A. The algorithm of FIG. 2 is an example of an algorithm to be used for the Mandelbrot Set. If a set of points from a different iterative domain is used in this method instead of the Mandelbrot Set, a different algorithmic function would instead be used for this different set of points. The algorithm starts (201), and a counter, n, is initialized to zero (step 221). A variable to be used in the algorithm, z.sub.0, is initialized to zero (step 227). A point p is chosen from region R, preferably the region corresponding to the area in and around the Mandelbrot Set (step 231). An example of choosing such a point might be to overlay a grid upon the Mandelbrot Set and then choose one of the points in the grid.

Detail Description Paragraph - DETX (7):

[0015] Determine if N numbers have been calculated which constitute the comparison string (step 241). In other words, check if n=N. If all the numbers of the comparison string have not yet been calculated (step 241), then the point p is used as input to the **iterative algorithm** z.sub.n+1=z.sub.n.sup.2+p (step 251). For example, the first iteration based on a point p is z.sub.1=z.sub.0.sup.2+p, or z.sub.1=0+p, or z.sub.1=p. Since p is a complex number of the form a+bi when decomposed into its real and imaginary parts, z.sub.2 takes the form z.sub.2=(a.sup.2+2i*a*b-b.sup.2)+a+bi or (a.sup.2-b.sup.2+a)+i(b*(2a+1)).

Detail Description Paragraph - DETX (9):

[0017] If the absolute value of z.sub.n+1 is equal to 2.0 or less, increment n by one (step 271) and check if N numbers have been calculated which constitute the comparison string (step 241). In other words, the **algorithm iterates** until n=N . If n<N, then perform the next iteration on point p (step 251). This next iteration will calculate the next number in the string of numbers comprising the comparison string. The process iterates until a string of variables, z.sub.1 through Z.sub.N can be produced that is of length N.

Detail Description Paragraph - DETX (10):

[0018] If n=N (step 241), then the **comparison string has been generated**. However, the numbers in the **comparison** string may need to be transformed to have values within a value set of interest (step 281). Suppose the numbers in the example target string representing gene expression ratios are real numbers between 0 and 10. If we wish to explore the similarities between the **comparison string and the target string** the value set of interest would be the real numbers between 0 and 10. The numbers of the **comparison** string may need to undergo some transformation to produce real numbers in this range. One way to produce such a real number is the function r=10.0*b/.vertline.z.sub.n.vertline.. This will produce real numbers r falling in the range between 0 and 10 for z.sub.n=a+bi. Provide the **comparison** string (step 291), and the algorithm ends (step 299).

Detail Description Paragraph - DETX (12):

[0020] If the **comparison** string does not meet the pre-scoring criteria (step 113), then the current **comparison** string is no longer under consideration. Another **comparison** string is instead provided (step 107). The new

comparison string is generated using the **iterative algorithm** of FIG. 2 on a new point p from region R.

Detail Description Paragraph - DETX (13):

[0021] If the comparison string meets the pre-scoring criteria (step 113), then scoring of the comparison string is performed (step 121). Scoring refers to some test of the **comparison string using the target string**. In the example of real numbers r falling in the range between 0 and 10 described above, the score could be the correlation coefficient between the **comparison string consisting of numbers r and the target string**. A simple example of scoring might be counting the number of one-to-one matches between the **comparison string and the target string** over some length L where L&lt;=N*, where N* is the length of the target string. Alternatively, a one-to-one **comparison between numbers in the comparison and target strings** may be performed for a non-contiguous number L of the numbers. For example, compare the second, fourth, and sixteenth numbers for a number L=3.

Detail Description Paragraph - DETX (15):

[0023] If it is determined that the point should not be marked (step 123), determine if a sufficient number of the M target strings have been checked for the point p (step 129). For instance, in our gene expression example, there may be several experiments or datasets that are being scored against each comparison string. If more of the M **target strings should be checked, the comparison string is scored against another of the M target strings** (step 121).

Detail Description Paragraph - DETX (16):

[0024] If a sufficient number of the M **target strings have been checked (step 129), determine if a sufficient numbers of points corresponding to comparison** strings have been checked (step 133). If more of the points corresponding to **comparison** strings should be checked, provide another **comparison** string (step 107). The new **comparison string is generated** using the **iterative algorithm** of FIG. 2 on a new point p from region R. The same M **target strings will then be used to score the new comparison** string.

Detail Description Paragraph - DETX (18):

[0026] **Target strings may be analyzed and/or compared** by examining, either visually or mathematically, their relative locations and/or absolute locations within the region R. When scoring similarity measures between the **comparison strings and the target strings,** target strings with greater similarity are generally mapped closer to each other based on Euclidean distance on the map.

This is because comparison strings with greater similarity are generally closer to each other on the map. However, this is not always true because the metrics involved are more complicated. For example, shading of points corresponding to **comparison strings with high scores for a given target string** represents a metric which shows similarity between this target string and others mapped in this shaded region. The target strings in this case, however, may not appear close together on the map or display but can be identified as being similar.

Detail Description Paragraph - DETX (19):

[0027] Continuing to FIG. 1B, determine whether points in region R should be marked based on their relative scores or properties compared to other points in region R (step 139). If it is determined that the points should be marked (step 139), mark the points (step 141). For example, one might wish to mark all the points whose score falls within 10% of the highest score of a chosen **target string, or mark points whose comparison** strings have the lowest or highest Shannon entropy for the region. When marking points, it may be determined that an entire subregion of the region has a large number of points that do not meet the relative score criteria or other properties. For example, this subregion may be part of a grid. This may be used to determine whether this subregion is of interest or not.

Detail Description Paragraph - DETX (20):

[0028] Once the decision has been made as to whether such points should be marked (step 139), determine if a subregion of R is of interest (step 143). If a subregion of R is of interest (step 143), then this subregion is examined with higher resolution, called "zooming" (step 147). The subregion of R replaces the previous region R in selecting region R (step 104 of FIG. 1A). **Comparison strings will be generated** from the new subregion of R and will be scored against the same set of M target strings originally provided. Points in a subregion of interest, which were previously unchecked, will be examined because the new region R is a higher resolution version of the subregion of interest. The points in the subregion will tend to produce a greater percentage of similar **comparison** strings to those previously examined in region R. If the subregion of interest is a high scoring region this will, in general, produce a greater percentage of high scores and some differences will emerge

to
produce higher scores or properties which are closer to some desired criteria.

Detail Description Paragraph - DETX (21):
[0029] After zooming (step 147) and before examining the subregion, the **target strings and comparison** strings may optionally be transformed to attempt
to improve the precision and resolution of the mapping and marking in the method. Suppose in the gene expression example, the target strings values, instead of real numbers from 0 to 10, were binned into 10 contiguous intervals, such that the first bin corresponds to real number values from 0 to 1, the second bin to real number values from 1 to 2, etc. Suppose these bins were labeled 0 to 9. The target string would then be a string of integers with values from 0 through 9. Suppose that a similar transformation was done on the transformed comparison strings. Suppose the method is performed and after zooming (step 147), the gene expression ratios and comparison strings are split into 20 such bins from 0 to 0.5, 0.5 to 1.0, etc. Thus, the target and comparison strings will be re-scaled before repeating the process in the new subregion (104 of FIG. 1A).

Detail Description Paragraph - DETX (24):
[0032] It should be apparent to one skilled in the art that this technique can be used to study the behavior of any (scoring) function that uses the **target strings and the comparison** strings as variables. Attempting to find the highest value of the similarity measure scoring function is a particular case of this. As such, this method could be used to attempt to optimize any scoring function, using a **target string or multiple target strings and comparison** strings as variables, to find the function's minima and maxima. In addition, each comparison string can simply be used alone as input into the variables of a scoring function for such a purpose.

Detail Description Paragraph - DETX (25):
[0033] It should be apparent to one skilled in the art that this method can be used for data compression. If the model of the **target string represented by a comparison string is sufficiently similar to the target string, and the coordinates of the point p corresponding to that comparison string can be represented in a more compact way then the target string,** then the target string can be replaced with its more compact representation in the form of the coordinates of point p. This is because the **comparison string generation** algorithm can then be used to recreate a sufficiently similar representation of target string from point p.

Claims Text - CLTX (2):

1. A method for manipulation, storage, modeling, visualization and quantification of datasets comprising: providing a plurality of target strings comprising datasets; **generating a comparison string** comprising a dataset using an **iterative algorithm, such that the comparison** string is calculated from a point in any set of points that can serve as the domain of an **iterative** function; scoring of the **comparison string by evaluating a function having the comparison string and one of the plurality of target strings** as inputs, such that the evaluation may be repeated for a number of the other plurality of target strings; mapping or marking the point if the score or some other property corresponding to the point meets some relevant criteria; repeating the generating, scoring, and mapping or marking for a plurality of **comparison** strings if desired; and examining a subregion with higher resolution if points in the subregion are of interest.

Claims Text - CLTX (6):

5. The method of claim 1, wherein the step of **generating the comparison string** comprises laying a grid over the set of points.

Claims Text - CLTX (7):

6. The method of claim 1, wherein the step of **generating the comparison string** comprises restarting the step of **generating the comparison string** if the iteration has become unbounded.

Claims Text - CLTX (8):

7. The method of claim 1, wherein the step of **generating the comparison string comprises generating a comparison string** of any length.

Claims Text - CLTX (10):

9. The method of claim 1, wherein the step of scoring comprises some test of the **comparison string using the target string**.

Claims Text - CLTX (11):

10. The method of claim 9, wherein not all of the numbers in the **comparison string or the target string** must be used in the test.

Claims Text - CLTX (12):

11. The method of claim 1, wherein the step of scoring comprises a one-to-one **comparison between corresponding numbers in the target string and the comparison** string.

Claims Text - CLTX (13):

12. The method of claim 11, wherein the one-to-one **comparison may be between corresponding sequential or non-sequential numbers in the target string and the comparison** string.

Claims Text - CLTX (16):

15. The method of claim 1, wherein the step of mapping or marking comprises storing the coordinates of the point corresponding to the **target string or properties of the comparison** string in memory, a database or a table.

Claims Text - CLTX (18):

17. The method of claim 1, wherein the criteria comprises the **comparison string having the highest score, where the score is based on some similarity measure to the target string**.

Claims Text - CLTX (30):

29. The method of claim 1, wherein separates processes involved in **generating each comparison string, scoring each comparison string, or transforming each comparison string or target string** to a value set of interest may be processed simultaneously by a plurality of processors.

Claims Text - CLTX (31):

30. A method for manipulation, storage, modeling, visualization and quantification of datasets comprising: providing a plurality of target strings comprising datasets; **generating a comparison string** comprising a dataset using an **iterative algorithm, such that the comparison** string is calculated from a

point in a region of the complex plane and the numbers of the **comparison** string are transformed to have values within a set of interest; scoring of the **comparison string by evaluating a function having the comparison string and one of the plurality of target strings** as inputs, such that the evaluation may be repeated for a number of the other plurality of target strings; mapping or marking the point if the score or some other property corresponding to the point meets some relevant criteria, such that the coordinates of the point corresponding to the **target string or properties of the comparison** string are stored in memory, a database or a table, or the point is marked on a visual display by changing some graphical property of the corresponding pixel, and wherein the relevant criteria comprises the **comparison string having the highest score, where the score is based on some similarity measure to the target string**; repeating the generating, scoring, and mapping or marking for a plurality of **comparison** strings if desired; and examining a subregion with higher resolution if points in the subregion are of interest, wherein the points of interest are analyzed and/or **compared** by examining, either visually or mathematically, their relative locations and/or absolute locations within the region or other metrics representing the graphic properties of the corresponding **comparison** strings.